ARITHMETIC LOGIC UNIT OVER FINITE FIELD GF(2^m)

Cross-References to Related Applications

This application is related to Korean Patent Application No. 10-2003-0007226 filed February 5, 2003, and takes priority from that date.

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates, in general, to arithmetic logic units over a finite field GF (2^m) and, more particularly, to an arithmetic logic unit, in which a division algorithm based on a binary greatest common divisor algorithm and a most significant bit-first multiplication algorithm share common logic such as common hardware logic, and both a multiplication and a division can be performed using the shared hardware device.

Description of the Related Art

As disclosed in Korean Pat. Appl. No. 1995-22327 (hereinafter referred to as "prior art"), in a conventional multiplication and division unit, a support circuit for multiplication and division operations includes first and second registers for storing input data, a first multiplexer for multiplexing outputs from the second register, an arithmetic logic unit for receiving outputs from the first register and the first multiplexer and arithmetically operating the received outputs in response to an input arithmetic control signal, a shift register capable of reading and writing signals in parallel so as to receive an output from the arithmetic logic unit, perform left and right shifting operations for a multiplication and a division and provide the arithmetic control signal, a gate connected to the arithmetic logic unit so as to gate a negative flag and an overflow flag and output the gated results, and a second multiplexer for receiving and multiplexing the output from the arithmetic logic unit, the output from the gate and the output from the first multiplexer.

However, the prior art is problematic in that the multiplication and division unit of the prior art is divided into structures for performing a multiplication and a division, respectively, and it is not possible to share a single hardware device and perform both a multiplication and a division using the shared hardware device, which are technical characteristics to be accomplished by the present invention.

SUMMARY OF THE INVENTION

Accordingly, the present invention has been made keeping in mind the above problems occurring in the prior art, and an object of the present invention is to provide an arithmetic logic unit, which has functions of performing both a multiplication and a division over a finite field GF(2^m) using a single hardware device.

By way of general background and as well known to those skilled in the art, arithmetic over the finite field GF(p), or Galois Field, can be useful for efficiently performing numeric calculations in computing devices. Because of its convenience in the context of binary computing devices, a finite field GF(2^m) can be selected. The finite field GF(2), referred to as the Galois Field of order 2, consists of the set of {0,1}. Accordingly, every element of GF(2^m) can be expressed as a polynomial having exponents between 0 and m-1, and coefficients that are either 0 or 1. With the selection of an irreducible polynomial associated with the finite field GF(2^m) for a given m, the coefficients associated with each polynomial term can be treated as a vector, and since the coefficients can only be zero or one, the coefficient vector can be treated as a binary integer. In this way, arithmetic operations can be carried out on the binary representations of the polynomials associated with the finite field GF(2^m).

In order to accomplish the above object, the present invention provides an arithmetic logic unit over a finite field GF(2^m) proposed to perform a multiplication algorithm of FIG. 1 and a division algorithm of FIG. 2. The arithmetic logic unit comprises a control logic unit, an RS-block unit, an SR-block unit and a UV-block unit, and has a function of performing both a multiplication and a division over the finite field GF(2^m).

BRIEF DESCRIPTION OF THE DRAWINGS

The above and other objects, features and other advantages of the present invention will be more clearly understood from the following detailed description taken in conjunction with the accompanying drawings, in which:

FIG. 1 is a view showing a Most Significant Bit (MSB)-first multiplication algorithm according to an embodiment of the present invention;

FIG. 2 is a view showing a division algorithm according to an embodiment of the present invention;

FIG. 3 is a block diagram of an arithmetic logic unit for performing both a multiplication and a division according to an embodiment of the present invention;

FIG. 4 is a circuit diagram of a control logic unit of FIG. 3;

FIG. 5 is a circuit diagram of an RS-block unit of FIG. 3;

FIG. 6 is a circuit diagram of an SR-block unit of FIG. 3; and

FIG. 7 is a circuit diagram of a UV-block unit of FIG. 3.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Hereinafter, embodiments of the present invention will be described in detail with reference to the attached drawings.

Reference now should be made to the drawings, in which the same reference numerals are used throughout the different drawings to designate the same or similar components.

FIG. 1 is a view showing a multiplication algorithm implemented according to the present invention, and FIG. 2 is a view showing a division algorithm implemented according to the present invention. The present invention implements a multiplier and a divider capable of executing the above algorithms, respectively, analyzes the structures of the multiplier and the divider, and recognizes, on the basis of the analyzed results, that a hardware device is shareable. The present invention combines the analyzed results to design an arithmetic logic unit having a function of performing both a multiplication and division over a finite field GF (2^m) using a single hardware device.

FIG. 3 is a block diagram of an arithmetic logic unit for performing both a multiplication and a division according to an embodiment of the present invention. The arithmetic logic unit includes a control logic unit 1, an RS-block unit 2, an SR-block unit 3 and a UV-block unit 4, which will be described in detail with reference to FIGS. 4 to 7.

The control logic unit 1 of FIG. 4 generates control signals required for the SR-block unit 3 and the UV-block unit 4 while outputting an externally-applied signal mult/div without change to be used as an input to select a multiplication or division operation.

That is, the control logic unit 1 generates the signal mult/div in response to an external control signal, and then outputs the signal mult/div to both the SR-block unit 3 and the UV-block unit 4, thus setting an operation of the arithmetic logic unit to a multiplication or a division.

Further, the control logic unit 1 generates control signals Ctrl1, Ctrl2, Ctrl3, state and c-flag used to control the RS-block unit 2, the SR-block unit 3 and the UV-

block unit 4 so as to perform the above multiplication or division operation of the arithmetic logic unit.

In this case, the control logic unit 1 includes one-bit registers, state and c-flag, an OR gate E1 and an XOR gate D1, as well as AND gates G1, G2, G3, G4 and G5.

The register c-flag is initialized to "1" when starting a division while operating together with the SR-block unit 3.

The AND gate G1 receives an output value state from the resister state, and also receives an output value b_i/z-flag from the SR-block unit 3 through an inverter.

The AND gate G2 receives an output value r_0 from the RS-block unit 2, and also receives the output value state from the register state through an inverter.

The AND gate G3 receives the output value state from the register state, and updates a value output from the register c-flag, when receiving the output value b_i/z-flag from the SR-block unit 3.

The AND gate G4 receives an output value r_0 from the RS-block unit 2 and also receives an output value a_0/v_0 from the UV-block unit 4.

The AND gate G5 receives the output value r_0 from the RS-block unit 2, and outputs the control signal Ctrl3 to the RS-block unit 3 when receiving the output value state from the register state through an inverter.

The OR gate E1 outputs a signal used to update the value, output from the register state, using the values output from the AND gates G1 and G2.

The XOR gate D1 outputs the control signal Ctrl2 to the UV-block unit 4 using the value output from the AND gate G4, and a value P_{m-1}/u_0 output from the UV-block unit 4.

The register c-flag outputs the control signal c-flag to the SR-block unit 3 using the value output from the AND gate G3.

The RS-block unit 2 of FIG. 5 performs an operation on R and S in the division algorithm of FIG. 2, and transmits the output value r_0 to the control logic unit 1.

That is, the RS-block unit 2 is constructed by arranging a plurality of circuits in cascade, in each of which one-bit registers r and s, an AND gate G6, an XOR gate D2 and a multiplexer MUX1 are connected to each other, so that, when the control signals Ctrl1 and Ctrl3 are received from the control logic unit 1, the output value r_0 is generated and output to the AND gates G2, G4 and G5 of the control logic unit 1.

That is, an output value r_1 from a register r_1 is input to both the XOR gate D2 and the multiplexer MUX1, which is constructed to receive a value s_1 output from the register s_1, and the control signal Ctrl3 output from the control logic unit 1.

In this case, an output value from the multiplexer MUX1 is input again to the register S_1 and then an output value from the register s_1 is input to one input terminal of the AND gate G6. The control signal Ctrl1, output from the control logic unit 1, is input to the other input terminal of the AND gate G6.

The register r_0 is constructed to generate the output value r_o, which is provided to the AND gates G2, G4 and G5 of the control logic unit 1, when the XOR gate D2 generates a new output value using the value output from the AND gate G6.

In FIG. 5, r_1,¼, r_{m-2} and r_{m-1} and s_2, ¼, s_{m-1} and s_m represent one-bit registers, and MUX1 represents 2-input multiplexers.

Meanwhile, FIG. 6 is a detailed circuit diagram of the SR-block unit 3. The SR-block unit 3 is constructed so that a plurality of one-bit registers b_{m-1}/sr_0, b_{m-2}/sr_1, ¼, b_1/sr_{m-2} and b_0/sr_{m-1} and two-input multiplexers MUX2, which are arranged in cascade, are each connected to one OR gate D3.

The OR gate D3 receives the signal mult/div from the control logic unit 1 through an inverter, and also receives the output value state from the register state of the control logic unit 1.

The multiplexers MUX2 output signals cnt_1, cnt_2, ¼, cnt_{m-1} and cnt_m used to update the values b_{m-1}/sr_0, b_{m-2}/sr_1, ¼, b_1/sr_{m-2} and b_0/sr_{m-1}, respectively, using the output value from the OR gate D3, the output value c-flag from the register c-flag of the control logic unit 1, and the output values b_{m-1}/sr_0, b_{m-2}/sr_1, ¼, b_1/sr_{m-2} and b_0/sr_{m-1}, which are fed back from the registers b_{m-1}/sr_0, b_{m-2}/sr_1, ¼, b_1/sr_{m-2} and b_0/sr_{m-1}, respectively.

After the registers b_{m-1}/sr_0, b_{m-2}/sr_1, ¼, b_1/sr_{m-2} and b_0/sr_{m-1} are constructed to update their output values using the signals cnt_1, cnt_2, ¼, cnt_{m-1} and cnt_m, which are output from the multiplexers MUX2, they feed back the updated values to the multiplexers MUX2, and to output the value b_i/z-flag to the AND gates G1 and G3 of the control logic unit 1.

In this case, the SR-block unit 3 uses m-bit bidirectional shift registers, instead of a $\log_2(m+1)$-bit counter, so as to implement a counter associated with the count value of the division algorithm of FIG. 2.

That is, if "0" (zero) is applied to the signal mult/div when the multiplication operation of FIG. 1 is performed, the values from the bidirectional registers shift in only a left direction because the state value is always "1" (one).

Further, if "1" is applied to the signal mult/div when the division operation is performed, the values from the bidirectional registers shift in left and right directions according to the state value.

FIG. 7 shows the UV-block unit 4 for performing an operation on U and V in the division algorithm of FIG. 2.

Referring to FIG. 7, the UV-block unit 4 is constructed so that a plurality of registers P_{m-1}/u_o, P_{m-2}/u_1,¼, P_1/u_{m-2} and P_0/u_{m-1} are connected in cascade so as to output a value P_{m-1}/u_0 to the XOR gate D1 of the control logic unit 1.

Further, in the UV-block unit 4, a plurality of registers a_0/v_0, a_{m-1}/v_1, ¼, a_2/v_{m-2} and a_1/v_{m-1} are connected in cascade so as to output a value a_0/v_0 to the AND gate G4 of the control logic unit 1.

Further, in the UV-block unit 4, multiplexers MUX3, AND gates G7 and G8, and XOR gates D4 and D5 are connected in cascade so as to update values output from the registers P_{m-1}/u_o, P_{m-2}/u_1, ¼, P_1/u_{m-2} and P_0/u_{m-1} and a_0/v_0, a_{m-1}/v_1, ¼, a_2/v_{m-2} and a_1/v_{m-1}.

Moreover, the UV-block unit 4 includes an AND gate G9 that consistently generates "0" in the multiplication mode to allow the multiplexers MUX3 to select the values output from the registers a_0/v_0, a_{m-1}/v_1, ¼, a_2/v_{m-2} and a_1/v_{m-1} in response to the signals mult/div and Ctrl3, which are output from the control logic unit 1, and an AND gate G10 that consistently generates "0" in the division mode.

That is, in FIG. 7, the control signal Ctrl2, the signal P_{m-1}/u_0, and the signal mult/div are input to one multiplexer MUX3. The control signal Ctrl1, the signal b_i/z-flag and the signal multi/div are input to another multiplexer MUX3. A value output from the former multiplexer MUX3 and a value g_{m-1}/g_1 are input to the AND gate G7. The value a_{m-1}/v_1 and a value output from the latter multiplexer MUX3 are input to the AND gate G8. A value output from the AND gate G8 and the value P_{m-2}/u_1 are input to the XOR gate D4. A value output from the AND gate G7 and a value output from the XOR gate D4 are input to the XOR gate D5 to allow a value output from the one-bit register P_{m-1}/u_0 to be updated, and then the value P_{m-1}/u_0 is output to the control logic unit 1.

Meanwhile, the signal mult/div and the control signal Ctrl3 are input to the AND gate G9. When an output value from the AND gate G9 and the output values P_{m-1}/u_0 and a_0/v_0 from the one-bit registers are input to the other multiplexer MUX3 to generate an output value, the output value is input to the one-bit register a_0/v_0. Therefore, the one-bit register a_0/v_0 outputs a value a_0/v_0 thereof to the control logic unit 1. The output value a_0/v_0 is re-input to the multiplexer MUX3.

In this case, the control signal mult/div is input to the AND gate G10 through an inverter, and the output value a_0/v_0 from the one-bit register a_0/v_0 is also input to the AND gate G10. The AND gate G10 consistently generates "0" in the division mode.

In this case, Table 1 compares the arithmetic logic unit of the present invention and a conventional multiplication and division unit.

Table 1. Performance of conventional dividers and arithmetic logic unit of present invention

	Brunner [1]	Guo [2]	Arithmetic unit of the present invention
Throughput (1/cycles)	1/2m	1/m	1/2m-1
Delay (cycles)	2m	5m-4	2m-1
Maximum processing delay	Tzero - detector + $2T_{AND2}$ + $2T_{XOR}$ + $2T_{MUX2}$	T_{AND2} + $3T_{XOR2}$ + T_{MUX2}	$2T_{AND2}$ + $3T_{XOR2}$ + T_{XOR2}
Components of circuit	AND_2: $3m+\log_2(m+1)$ XOR_2: $3m+\log_2(m+1)$ Latch: $4m+\log_2(m+1)$ MUX_2: 8m	AND_2: 16m - 16 XOR_2: 10m - 10 Latch: 44m - 43 MUX_2: 22m – 22	AND_2: 3m + 7 XOR_2: 3m + 1 OR_2: 2 Latch: 5m + 2 MUX_2: 3m + 2 Inverter: 5
The number of transistors	$110m + 18\log_2(m+1)$	608m – 432	88m + 84
Operation	Division	Division	Multiplication/division

AND_i: i-input AND gate,

XOR_i: i-input XOR gate,

OR_i: i-input OR gate,

MUX_i: i-to-1 multiplexer,

T_{ANDi}: transmission delay generated through one AND_i gate,

T_{XORi}: transmission delay generated through one XOR_i gate,

T_{MUXi}: transmission delay generated through one MUX_i gate, and

Tzero-detector: transmission delay generated through $\log_2(m+1)$-bit zero-detector.

As described above, the present invention provides an arithmetic logic unit over a finite field GF (2^m), which reduces a calculation delay and the number of transistors used to implement a required hardware device by comparing and analyzing only a divider function of the arithmetic logic unit of the present invention and those of the conventional dividers, as shown in the above Table 1.

Further, in the prior art, separate multiplication and division modules were used to implement an arithmetic logic unit over a finite field GF(2^m). However, the present invention does not require separate multiplication and division modules by utilizing

shared logic resources in the arithmetic logic unit.

Therefore, the arithmetic logic unit of the present invention is very suitable to implement an encryption system of applications requiring a small area, such as smart cards or wireless communication devices. Further, since the present invention has high expansibility and flexibility with respect to the size m of a field, it can be variously applied to arithmetic logic units over the finite field GF (2^m), and it is very useful for industries using an encryption system.

Although the preferred embodiments of the present invention have been disclosed for illustrative purposes, those skilled in the art will appreciate that various modifications, additions and substitutions are possible, without departing from the scope and spirit of the invention as disclosed in the accompanying claims.